

SINC - LINK

Vol. 6 No. 6

NOV-DEC '88

SINC-LINK IS A PUBLICATION OF THE TORONTO TIMEX-SINCLAIR USERS CLUB. COPIES OF THE NEWSLETTER ARE \$1.50 EACH FOR NON-MEMBERS. CLUB MEMBERS RECEIVE A FREE COPY AS PART OF THE \$20.00 ANNUAL MEMBERSHIP FEE.

NEWSLETTERS ARE EXCHANGED, FREE OF CHARGE, WITH OTHER TIMEX-SINCLAIR USER GROUPS.

ALL MATERIAL IS PUBLIC DOMAIN AND CAN BE REPRINTED. PLEASE CREDIT THIS PUBLICATION AND THE AUTHOR IF YOU COPY MATERIAL.

SEND CORRESPONDANCE TO:

Attention: SINC-LINK EDITORS,
TORONTO TIMEX-SINCLAIR USERS
CLUB, 14 RICHOME COURT,
SCARBOROUGH, ONTARIO, CANADA,
M1K 2Y1

LATE FALL EDITION

Page 2	EDITORIAL
Page 3	MScript 5.3 REVIEW
Page 4,5	MEMBERS' LETTERS
Page 6	LOTTERY PROGRAM
Page 7	QL BEGINNERS
Page 8	"HACKER" REVIEW
Page 9	LARKEN TIPS
Page 10	PROJECTS
Page 11	LARKEN LIBRARY?
Page 12	"FIRST CLASS FONTS"
Page 13	TASWORD DRIVER
Page 14	THE LAST PAGE

EXECUTIVE OFFICERS:

PRESIDENT:
TREASURER:
SECRETARY:
ACTIVITIES DIRECTOR:
TAPE LIBRARIAN ZX81:
TAPE LIBRARIAN 2068:
PAPER LIBRARIAN:
NEWSLETTER:
LIAISON OFFICER:
(Out-of-town members)

JEFF TAYLOR (244-8583)
BILL LAWSON (444-8772)
GEORGE CHAMBERS (751-7559)
RENE BRUNEAU (531-9749)
" " "
RENATO ZANNESE (635-6536)
GREG ROBINS (920-7747)
JEFF TAYLOR
GEORGE CHAMBERS, 14 RICHOME COURT,
SCARBOROUGH, ONTARIO, M1K 2Y1
(416-751-7559)

TORONTO TIMEX - SINCLAIR USERS CLUB

14 RICHOME COURT, SCARBOROUGH, ONTARIO, M1K 2Y1
CANADA

Editorial

I could weep! All that time wasted waiting for tapes to load and then having them fail, forcing me to reload again and again. That's it, finished, finito, never again.

No, I haven't given up on the old 2068 but I have given up fighting a marginal mass storage medium, namely tape.

After 5 years of frustration, boredom and hair-pulling I finally broke down and bought a disk drive system. Larken's to be specific. From the second I ran Larry Kenny's system disk, I cursed myself for not ordering this package sooner. My initial resistance, based on whether I could justify the price of a system costing twice as much as my computer, evaporated as soon as I saved a program to disk then loaded it back in. The time savings is staggering! MSCRIPT tape - 90 seconds, MSCRIPT disk - 5 seconds. When I calculate all the time I have spent staring at walls, while waiting for tapes to load/save, it works out not to minutes or hours but to days and since I have almost no patience this system is a godsend. I won't go into anymore details except to say that I just love that little NMI button!

A few members helped me get my system running. Special thanks to George Chambers for trouble-shooting and aligning my used drives, to Renato Zannese for assembling a really neat little power supply and to Bob Mitchell for supplying an interface cable late one night on very short notice. Also my apologies to Larry Kenny for pestering him to send me my system once I had ordered it. As I mentioned before, I hate to wait for anything. 'Nuff said!

Computer Fest '88 was held mid-October and once again George Chambers and Rene Bruneau took to the Timex-Sinclair experts' chair. Visitor response was lukewarm but a few remembered their old ZX81's and maybe our brochure will revive their interest. Once again, thanks go to our experts for donating their time.

Club Executive Officer elections took place at the October meeting, see the newsletter cover for new and returning positions. I must say that the members' choice for president showed exceptional good taste but then I may have a slight personal bias on the subject. I hope he can live up to expectations.

The newsletter format is changing in that we are going to try to print a new hardware project every issue. The cover will be changing too. It's time for some artwork. Any suggestions? Any articles? Any questions? We're here for you, so let us know what you want. (We're reasonable - sort of).

One last point. Be aware that we are changing our mailing address. Do not use the post office box anymore. It will be closed in the next couple of months. Use the address shown on the cover.

Keep on Sinc'ing (no pun)

J.T.

FOR SALE FOR SALE FOR SALE

- 1 n TS 2068 w/Power Supply (has RGB monitor output).
- 1 n Larken 4D System
(Early version- w/cartridge & Spectrum Eprom.
Uncased & heavily modified for 3 drives)
- 1 n OS64 Eprom only (can be used with Larken Cartridge)
- 1 n TS2050 modem (uncased)
- 1 n TS 2040 Printer w/2 rolls of paper
- 1 n Compudeck JT1115 Tape Recorder
- 1 n Printer interface (connects to Larken I/F)
- 1 n Pascal manual
- 1 n Book of Spectrum programs
- 1 n Zeus Assembler manual plus original tape
- 1 n Technical manual for TS2068 (complete w/appendx's)
- 2 n Tapes of programs
- 1 n Diskettes
- 1 n Package of documentation for programs on disks

Complete package \$200

For further information contact:

Luca Martini

6 Maplewood,

Outremont, Que H2V 2L8

FOR SALE FOR SALE

- 1 n TS 2068
- 1 n TS 1500
- 1 n TS 1000
- 1 n Parallel Printer I/F
- 1 n OS 64 Cartridge Board
- 1 n GE Tape Recorder

Write for more information to:

Al Gedris, 355 Royal Oaks Blvd.,
Richmond Heights, OH 44143-1709

My long-awaited copy of Version 5.3 of MSCRIPT by Jack Dohany finally arrived early in June of this year. Because I had told him that I was using a non-standard, earlier version of the Larken DOS, he sent the program on cassette, to be on the safe side. I was delighted to find that a back-up on disk was readily made by selecting DISK on the main menu and then pressing B.

Documentation was on paper and the only other program on the cassette was the LCODES text which contains all the various codes that a wide printer would recognize. I understand that the disk version contains all the documentation as files.

The main problem that I had had with my earlier version MS5T was one of the printer dropping characters every now and then. This problem has been resolved, I am happy to report.

The cost to me for the new version was US\$25 which included \$5 for mailing and handling. The program is so much better than the earlier version that I was happy to pay the price.

Here are some features of MSCRIPT (not necessarily in order of importance) that make the program quite attractive and make me a convert:

1. Finding and Changing any string of up to 30 characters is simple and almost instantaneous.

2. Headers and Footers (titles at the top and bottom of each page of a document) are easy to add.

3. Moving, copying, deleting, saving and loading blocks of text is fast, easy and very professional, with markers used to identify the block to be worked on.

4. The cursor moves very fast across the page. There is also a tab feature that is very useful when developing tables and lists.

5. When making a back-up copy, it is optional to include the text as part of the back-up; ideal for making a version that includes, say, a letter head for page 1. If you want to make a LARKEN AUTOSTART copy, this can be done by using <CLEAR 45000;RANDOMIZE USR 102;RUN>. You have to CLEAR a higher address if you include any text in your save.

6. View Memory allows you to PEEK and POKE the contents of memory; this starts at the LCODE file which is handy if you want to inspect and change any printer control codes.

7. The name of the text you are working on appears on the home menu. I find this to be very useful particularly as I hop from text to text fairly often. It let's me know where I am at the moment. The GIVE NAME command allows you to assign or change the name of your text at any time.

8. There is an alternative fattened character set that makes reading the text on my monitor much easier. The use of different screen colours is restricted by the TS2068 hardware and I have settled on white ink on black paper as the best for me.

9. The disk catalogue in the selected drive can be added to the text in progress. The drive is selected by entering a number from 0 to 3. The LARKEN Ramdisk is selected with <4> and this can be accommodated easily by changing lines 131 and 132 in the BASIC with no addition to the BASIC length.

10. The new LARKEN Version 3 EPROM uses the token MOVE to call a file rename routine and this was easily added to the home menu by a few lines at line 220 in the BASIC. Watch out for an out of memory report here. I had to sacrifice lines 420 to 440 (VERIFY, which I seldom use anyway) to make room for this.

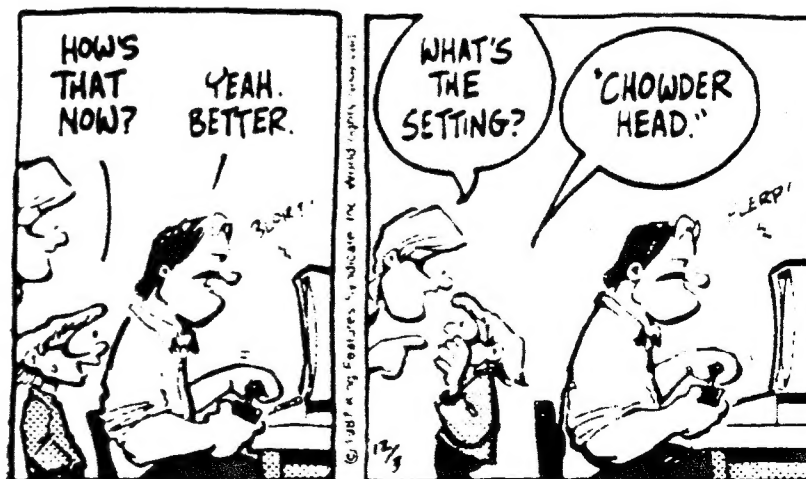
11. I like being able to put comment lines in the text to serve as reminders to me and that won't appear in the printout. Tab settings are one example as you have to reset these whenever they default to their normal settings.

One thing that could be better: It's too easy to stutter in more spaces and letters than wanted when entering text.

MSCRIP is a totally different word processor from TASWORD which I have used for some time and which has its devotees. But there are so many advantages to MSCRIPT that I have begun to switch over to it for my word processing tasks. Once you get used to the protocol and press the right keys, MSCRIPT is just as easy to use as TASWORD and a lot more rewarding. I recommend the purchase of MSCRIPT Version 5.3 and look forward to experimenting with Version 6 when it comes along. Hurry up, Jack!

307

On The Fastrack



LETTERS FROM OUR MEMBERS
Topics...Larken & Quad drives, Programming, SIPs,
Sequential Filing, Deeks, etc.
by George Chambers

As liaison to our out of town members I often get letters which contain stuff that is just too good to be filed away. Typical are two letters that I have received recently. I have selected sections from them which seem likely to be of interest to other members.

Richard Hurd, Seaside, Oregon has a Larken system with a pair of Amdek 3 inch drives and a Tandon quad drive. He writes as follows:

".....Adding the new Tandon FD, brought out new problems (such a harsh word) with it "fitting in" to the system. I've put together a few utilities that I call "ODDBALL" to handle a few of them. It is compatible with all LKDOS cartridge owners. I've enclosed a copy, for you and the groups' collection. But I need to ask a favor of you before you pass out any copies. Please test out the Disk Conversion and Disk Copy using your Double Sided, Double Density FD's. It hasn't been tested with that combination. It has a HELP file to explain itself (hopefully).

More from Richard: "...After seeing the group's newsletter I am excited about getting the new eprom. Remember I said I had placed an order for Larry Kenny's Sequential Filing utility? (and now I hear that the new EPROM has built in Sequential File Handling; wonderful!) At the time I also asked a question about Quad Density drives and the possibility of reading/ writing Double Density format. Well, he wrote back and explained how to read double density disks with the quad density drive. Track 0 is the first track on the first side, compatible with both drives. But, after track 0, the QD drive moves two tracks to the DD drive's 1 track move. Like::

80 TRACK (QD) 40 TRACK (DD)

```
0 000000000000 0
1
2 000000000000 1
3
4 000000000000 2
5
6 000000000000 3
7
8 000000000000 4
etc.
```

"....For example, the directory looks at the blocks used by a file and then sets the drive's head to that track. But LKDOS LOAD won't load a track whose block used marker (byte 2 in a block) doesn't match CURTRK. One way around it is to use the cartridge routines TRACK and LOADBF. A short routine to copy the appropriate tracks off of a Double Density disk and then save them to a Quad Density disk all using one Quad Density drive should do it."

End of quotes from Richard's letter.

In a letter from another correspondent, Steven Gunhouse, Livonia, MI (USA). I had mentioned the possibility that I needed a SIP in my 2068 to run some Spectrum programs::

"...A word about the thing of attaching a SIP (Single Inline Plastic package) resistor network. It may not have been necessary for your computer. I know it wasn't for mine.

"Most programs wouldn't need it anyway. The only ones which require a resistor network are those that work on the basis of interrupts. The only one that I am familiar with is EZEDIT. In fact that may be the place where you got the idea of adding a SIP. On my computer however, EZEDIT ran without changing the hardware.

"On an unmodified 2068 with no extras, Timex did not put pull down or pull up resistors on all the data lines. That means that if you are in a condition when neither memory nor Z80 are putting stuff onto the bus, you have no way of knowing what condition the data lines are in. However, either the AAERCO printer interface or disk interface on my system must include such resistors. Additionally, if you have a Spectrum emulator, which is any more than a ROM, it probably has them as well.

"There is one easy way to find out. All you have to do is a machine-level IN command from a non-existent port. Something such as PRINT IN 0 should be sufficient. If you get anything other than 255, your machine does not have pull-ups and can benefit from a SIP properly installed or you do not have a port 0 (the number zero).

"...I have thought about your "DEEK" function. ("DEEK" = Double Peek, gfc) There are only about a half a dozen ways to do it. The choice depends mostly on how you would like to pass your parameters. If the USR function allowed a second parameter, then all would be easy. You would just write a simple ml program and then type PRINT USR (DEEK, a) and that would be that. Of course, DEEK and "a" would be numbers, the location of your ml, and the address to DEEK.

"However things are not that easy. So you will have to find a method to pass the address to the ml. Also, you will have to decide whether to pass the result to BASIC, or do something else with it. Oh, you could also DEF FN in BASIC to get your DEEK, as for example Tasword II does, with the line:

```
1 DEF FN d(a)= PEEK a+256* PEEK (A+1)
2 PRINT FN d(address): REM To show how it
would be used.
```

"The obvious methods in BASIC are: Store the address in memory with 2 pokes or RANDOMIZE, find the variable, store it as a FN parameter, get it from the math stack, or do something to actually change BASIC. Obviously the RANDOMIZE approach is the easiest, if you don't need to use RND anywhere. The using two pokes is simulating DOKE in BASIC. It would look like this:

```
10 POKE address, INT (value/256): POKE
address+1, value, 256*PEEK address
```

For either of these approaches, the ml would simply be:

```
LD HL, (address)    Note: if you are using
LD B, (HL)           RANDOMIZE, "address" is
INC HL              SEED = 5C76, or 23670 dec.
LD C, (HL)
RET
```

The other methods are much more complicated. I can refer you to several articles in ZX COMPUTING Monthly for these, especially the Machine Code Calculator article in the October 1986 issue. As far as modifying BASIC goes, you could break into the routine at 0008 and modify the syntax of the PEEK command, but that is too complicated to go into here.

"...I have a rather unique use for sequential files to suggest. You can do something called programmed input, which is putting things that would normally be input into the file. You can do

this even with BASIC commands. So you could write a "program" on the disk, and then have the computer read from the disk what it thinks it is getting from direct input.

As a simple example, OPEN a sequential file, and LIST a program into it. Then print a CLOSE #0 to the disk, and close the channel. This effectively saves the program, though without the numeric "slugs" and such. Then you could later load it, not with a LOAD, but by opening stream 0 to the file. Actually this would be more like a MERGE, unless you included a NEW in the file before the listing.

What you would type is this: (with a program already in the 2068)

```
PRINT #4: open #5, "Program"
LIST #5
PRINT #5, CHR$ 245+"#4:"+CHR$ 212+"0"
PRINT #4: CLOSE #5
NEW: REM To erase the program
PRINT #4: OPEN #0, "Program": REM will rewrite
the entire program
```

You could do anything this way, just as you would from a keyboard. Note, the CHR\$ 245 is a PRINT, the CHR\$ 212 is a CLOSE #.

My only remaining complaint with the Larken DOS is the requisite PRINT #4. This could be gotten around as the British Interface One did, by breaking into the error routine at 0008, or by several other methods. However, it is certainly no worse than AERCO's non-standard commands - CAT to LOAD, MOVE to SAVE, etc.

Well, I hope this information has been useful. I am currently working fulltime....."
End of quotes from Steven's letter.

Does this prompt any other members to respond. Please write to me.

A DATA ACQUISITION SYSTEM FOR THE QL from Prairie Digital

We have received a one page flyer advertising a data acquisition system for the QL computer.

The sheet is a little bit too much to include in the newsletter, but I should like to give a few details about it.

It is a printed circuit board that plugs into the expansion connector of the QL.

The advert makes the following statements:

- 24 lines of programmable I/O
- 8 channel - 8 bit A/D converter
- 12 bit counter
- Comes with easy to follow instructions

The price is \$89.00 US for 1 system.

There are quantity discounts.

The bare board sells for \$30, and includes complete instructions

Add \$3 for shipping and handling...cheque, M.O. or C.O.D.

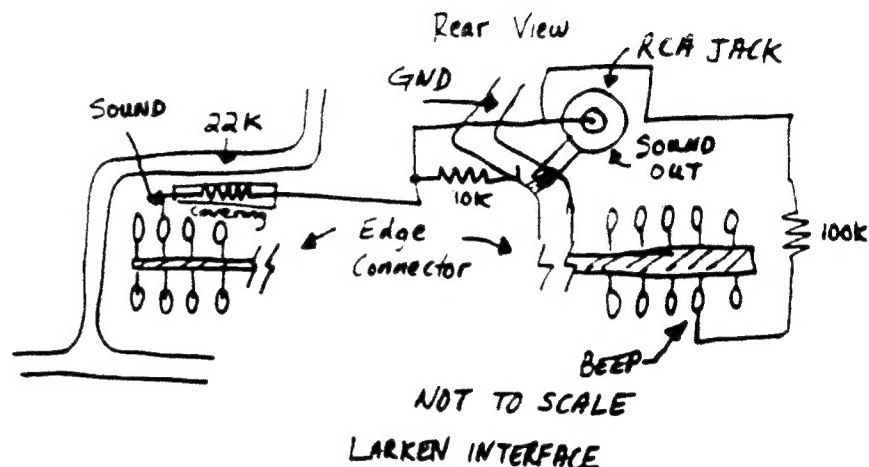
PRAIRIE DIGITAL 4561 SANDPIPER TRAIL,
COTTAGE GROVE, WI 53527 U.S.A.

I have not heard about these people before, so I can give no particular recommendation. You are on your own. Let us know how you make out. How about a review.

George Chambers

Error in Simple Sound Port

There seems to be an error in the drawing that I provided in the Sept - Oct 88 issue of Sinc-Link. Below you will find a new drawing.



Here is a simple program which can be used to pick lottery numbers. The program has been modified so it prints the numbers to a 80 column printer automatically. The program can be used with a 2040 printer by modifying all LPRINT CHR\$ 1 to COPY. To save this Program go Line 9985. Good luck with this program. R. Zannese Oct., 23, 88

Con't next page

5 REM Modified for the Larken
Disk System By R. Zannese Oct., 2
3, 1988

10 BORDER 5: INK 9: PAPER 0: C
LS

15 PRINT #4: OPEN #3, "LP"
20 PRINT AT 1,8: " Lytle Softwa
re "; TAB 8; " LOTTERY "; TA
B 8; " Orignal "; TAB 8; "
MAR. 20, 1985 "; TAB 8; " Progr
ams "

30 PRINT AT 8,6: "1. LOTTO"; AT
10,6: "2. PICK FOUR"; AT 12,6: "3.
THE DAILY NUMBER"

35 PRINT AT 14,6: "4. 6-49"
40 PRINT AT 21,7: BRIGHT 0: "En
ter your choice"

50 PAUSE 0: LET J=VAL INKEY\$:
IF J<1 OR J>4 THEN GO TO 50
60 CLS : BORDER J: GO TO 1000+

J
1000 PRINT INK 7: PAPER 0: AT 0,
10: " L O T T O "

1010 PRINT : RESTORE
1020 LET b1=0
1030 FOR c=1 TO 5: READ a\$
1040 DATA "A", "B", "C", "D", "E", "F",
"G", "H", "I", "J"
1050 PRINT TAB 10: "Selection No.
"; a\$

1060 FOR a=1 TO 6
1070 LET b=INT (RND*39)+1
1080 IF b=b1 THEN GO TO 1070
1100 IF b<10 THEN PRINT " ";
1110 PRINT " "; b;
1115 LPRINT " "; b;
1120 LET b1=b: NEXT a
1130 PRINT TAB 3: "-----"

1135 LPRINT TAB 3: "-----"

1140 PRINT
1145 LPRINT
1150 NEXT c
1160 INPUT "C=copy R=reset E
NTER=next "; LINE d\$: IF d\$="c
" THEN LPRINT CHR\$ 1
1170 IF d\$="r" THEN CLS : GO TO
1000

1180 IF c=6 THEN CLS : PRINT ""
: FOR c=6 TO 10: READ a\$: GO TO
1050
1190 PRINT TAB 6: INK 2: PAPER 7
: FLASH 1: " G O O D L U C K "
1200 PAUSE 200: GO TO 10
2000 PRINT AT 6,7: " P I C K F O
U R "

2010 FOR c=1 TO 4
2020 LET p4=INT (RND*10)
2030 PRINT AT 12,8+(c*3): p4
2040 NEXT c
2050 PRINT AT 21,7: " Push "; FLA
SH 1: p4: FLASH 0: " for MENU "

2060 IF INKEY\$<>STR\$ p4 THEN GO
TO 2060

2070 GO TO 10
3000 PRINT AT 6,6: " T H E N U M
B E R "

3010 FOR c=1 TO 3
3020 LET p3=INT (RND*10)
3030 PRINT AT 12,10+(c*3): p3
3040 NEXT c
3050 PRINT AT 21,7: BRIGHT 1: " P
ush "; FLASH 1: p3: FLASH 0: " for
MENU "

3060 IF INKEY\$<>STR\$ p3 THEN GO
TO 3060

3070 GO TO 10
4000 BORDER 1: PRINT INK 7: PAP
ER 0: AT 0,10: " 6 - 4 9 "

4010 PRINT : RESTORE
4020 LET b1=0

4030 FOR c=1 TO 5: READ a\$
4040 DATA "A", "B", "C", "D", "E", "F",
"G", "H", "I", "J"

4050 PRINT TAB 10: "Selection No.
"; a\$

4055 LPRINT TAB 10: "Selection No
."; a\$

4060 FOR a=1 TO 6
4070 LET b=INT (RND*49)+1
4080 IF b=b1 THEN GO TO 4070
4100 IF b<10 THEN PRINT " ";
4105 IF b<10 THEN LPRINT " ";
4110 PRINT " "; b;
4115 LPRINT " "; b;
4120 LET b1=b: NEXT a
4130 PRINT TAB 3: "-----"

4135 LPRINT TAB 3: "-----"

4140 PRINT
4145 LPRINT
4150 NEXT c

4160 INPUT "C=copy R=reset E
NTER=next "; LINE d\$: IF d\$="c
" THEN LPRINT CHR\$ 1

4170 IF d\$="r" THEN CLS : GO TO
4000

4180 IF c=6 THEN CLS : PRINT ""
: FOR c=6 TO 10: READ a\$: GO TO
4050

4190 PRINT TAB 6: INK 2: PAPER 7
: FLASH 1: " G O O D L U C K "

4200 PAUSE 200: GO TO 10
9980 STOP

9985 REM larken save
9995 PRINT #4: SAVE "Lotto.B1" L
INE 10

BEGINNERS SECTION By Bill Lawson

THE FIRST REQUIREMENT OF ANY WOULD BE PROGRAMMER IS AN EFFECTIVE 'INPPUT' ROUTINE. IT IS POSSIBLE TO USE AN INPUT STATEMENT AND CHECK THE NATURE OF THE INPUT AGAINST WHAT IS REQUIRED OR ALLOWED IN ANY PARTICULAR CIRCUMSTANCES BUT FIRSTLY THE CHECKS ARE MUCH MORE CLUMSY TO CONSTRUCT AND SECONDLY IT PREVENTS ANY DIRECT CONTROL ON THE LENGTH OF THE INPUT STRING AND ALWAYS REQUIRES THE EXTRA KEY PRESS OF THE ENTER KEY.

IN PRCTICE IT IS MUCH MORE SATISFACTORY TO ESTABLISH THE VALID INPUT AND ONLY ENABLE VALID KEYS TO FUNCTION. THIS IS READILY ACHIEVED WITH THE 'INKEY\$' FORM OF INPUT.

THE REQUIREMENT OF AN INPUT ROUTINE INCLUDES:-

- 1) A SCREEN PROMPT
- 2) USUALLY, BUT NOT ALWAYS, A CURSOR TO INDICATE WHERE THE INPUT WILL OCCOUR.
- 3) A MEANS OF ENABLING ONLY RELAVANT KEYS.
- 4) CONTROL OF THE MAXIMUM AND MINIMUM NUMBER OF CHARACTERS TO BE ACCEPTED.
- 5) ASSIGNMENT OF THE INPUT TO A STRING OR NUMERIC VARIABLE AS APPROPRIATE.

PRCTICALLY ALL BUSINESS PROGRAMS IT WILL ALSO BE NECESSARY TO CREATE A STRING ARRAY CONTAINING THE PROMPTS AND THE RESPONSES TO INCLUDE IN A HARD COPY OUTPUT ROUTINE AND ALSO TO PROVIDE A DEFAULT RESPONSE, OFTEN THE RESPONSE FROM A PREVIOUS RUN OF THE PROGRAM ELEMENT.

SUITABLE ROUTINE FOR THE NEEDS OF A PARTICULAR PROGRAM CAN USUALLY BE WRITTEN QUITE BRIEFLY BUT IT IS PRFERABLE TO WRITE A ROUTINE WHICH HAS THE WIDEST POSSIBLE AREA OF APPLICATION SO THAT THE SAME CAN BE USED VIRTUALLY IN ANY PROGRAM.

WE WILL START AT THE SHARP END WITH THE PROCEDURE USED TO RECIEVE EACH AND EVERY KEY STROKE REQUIRED IN THE PROGRAM. THERE IS NO NEED FOR ANYY OTHER 'INKEY\$' STATEMENTS NO MATTER HOW LONG OR VARIED THE PROGRAM REQUIREMENT. THE FORM I RECOMMEND IS:-

```
DEFine PROCedure KEYINPUT(POS)
REPeat KEY
Z$=INKEY$(-10:POS=Z$ INSTR POS:IF POS:RETURN:ELSE:BEEP 2000,20
END REPeat KEY
END DEFine KEYINPUT
```

THE PROCedure IS CALLED WITH A STRING PARAMETER WHICH MUST CONTAIN ALL PERMISSABLE KEYS. IT WAITS FOR A KEY TO BE PRESSED. CHECKS WHETHEV THE KEY IS A VALID CHARACTER, RESPONDS WITH A BEEP IF NOT AND RETURNS THE CHARACTER IN VARIABLE Z\$.

COPIED FROM QUANTA NEWS/L V. 2 ISS. 10 NOV. 85

WRITTEN BY JOHN TAMMER 23 NORTHUMBRIA DR BRISTOL BS9 4HL (0272) 623401.

Con't next issue

HACKER
A program review
by Greg Robins

Late one night as you scan a number of electronic bulletin boards you try a certian number. You find your screen has suddenly turned blue. Thinking of a possible malfunction in your equipment you reach over to shut the computer off. It is then you notice written in white letters across the screen "LOG ON PLEASE".

Ah, there is no main screen advertising this program, no list of HELP keys as a clue as to where you are. There is nothing but the message. After two attempts at the password (in brackets) you notice a written "PRESS H FOR HELP". Doing so, you see printed "PASSWORD HAS BEEN CHANGED...NEW PASSWORD LOCATION TESTSITE...LOGON PLEASE". You have just started playing HACKER.

HACKER was originally released by Activision in 1985, followed 2 years later by it's sequel, HACKER II. I recently was able to purchase the original program from England. The tape came with no instructions to start you off, only with steps on how to load the program.

You start off logging onto a high security network. Due to a malfunction in the security you are connected. Then you receive a message giving you a clue as to what you have stumbled onto. Next you are presented with a diagram of a subterranean remote unit (SRU). You must then do a laser alignment test over all the droid's circuitry. This takes a number of tries, because if you make one mistake, you will receive a "test error" report, and have to start from the beginning.

Once this is successful you then are presented with a map of the world on the lower half of your screen, and the controls for the droid at the top half. Next you are asked to set the time of day you wish to begin. Once this is completed your mission begins. You must use your SRU to recover shredded pieces of the document scattered across the world. These pieces hold vital information to a project Washington wishes to learn more about. This project could mean world domination. So it's up to you to recover the information. Using your SRU, you travel the continents searching out spies who have the pieces of the document.

You end up in France, England, Russia, or anywhere on the globe. To get these important shreds of the document you must buy them or trade with these agents. WARNING! Some agents will take what another agent will want for his piece of the document. This is where things can get tricky. I was at first disappointed in this aspect of the game, because I felt that HACKER would be travelling through different networks, using information, telephone numbers, etc. to learn about the project. Until suddenly, I was caught up in dealing with the agents, trying to put the pieces of the document together, and to solve it before obtaining the last piece.

Once you have all pieces of the document you head to Washington where agent Levi will give you a password so you can log-on. Then the real hacking begins.

I have at the time of this review recovered four-fifths of the document. Soon I hope to have all the pieces so I can do what I really enjoy, especially in a game, hacking.

They say everyone, even if they will not admit it openly, enjoy doing something that is slightly illegal. With HACKER you do just that. Even if it is only in an artificial world.

And an excerpt from another member, Larry Crawford, London Ontario, who writes:

"....Thanks for your 30 Aug letter. I have been plagued with disk drive problems. Tried replacing everything everything else that I had not replaced before, with the same frustrating results. It finally dawned on me that it all worked smoothly when the drives were on the bench, but failed when put back in place on the shelf. Tried exotic shielding, rerouting the cable, etc. with no success. That wasn't the problem. The drives were sitting vertically on the bench and horizontally on the shelf. They are now humming away happily on the shelf, in the vertical position. I won't even speculate as to the actual problem, but as long as they keep working, I'm not going to mess with them any more."

AN UNUSUAL DISK DRIVE PROBLEM

One of our members, Bob Mitchell, a frequent contributor to our newsletter, recently ordered and received a quad drive from Ed Grey Enterprises. During the process of setting it up he encountered CRC errors. The errors occurred only in the high-numbered tracks. Tried the drive in another system; same result. Visual inspection of the drive showed that the head movement was sluggish toward the end of its travel, and when moved manually, sure enough there was some resistance to travel at the inner part of the disk. Very puzzling, until it was noticed that the lead from the drive head assembly, leading to the drive electronics, was snagging on the interface connector at the rear of the drive, preventing the head from it's full travel. Relocating the wires cleared the problem completely. G.F.C.

GOT ANY GAME TIPS, PEEKS OR POKES?

SEND THEM TO SINC-LINK AND WE'LL

PUBLISH THEM IN THE NEXT ISSUE AND

WE'LL CREDIT YOUR NAME!

In writing utilities for the Larken system I have developed a number of short routines which I incorporate into many of them. This article is to share some of them.

I like to use "buried" colour codes in my programs. I call them "buried" for lack of a better term. Although many of you are familiar with use of this programming trick, I shall expand on it for benefit of the uninitiated.

Type in the following line.

```
10 PRINT "GEORGE"
```

Bring the line down for editing, and move the cursor over to between the first quotation mark and the G. Now get into the "E" mode by pressing the CAPS SHIFT and SYMBOL SHIFT keys simultaneously. Then press one of the keys 1 to 7, or 0. The line being edited will change to the colour represented by the key you just pressed.

That's fine, but the colour will continue through the balance of your listing. You can remove it by moving the cursor over to say, between the letter E and the second quotation mark. Here you should again get into the "E" mode, then press the 7 key. This will bring the line being edited, back to the normal white PAPER colour.

This is all very well, but when one is listing a program during a debugging session, these "buried" colour codes mess the screen up terribly. This is why many programmers detest this programming gimmick.

What this is leading up to is the following line which I insert into my programs. Whenever I am writing a program I install a line 8000 as shown below. Then during the debugging process I enter GOTO 8000 to clear out the listing.

```
8000>INK 0:PAPER 7:BORDER 7:CLS :LIST
8999 STOP
```

Although you may think that having a disk system means never using a tape recorder again, 'tis not so. I use the tape on many occasions. This is why I have found it useful to incorporate the following lines into many of my programs. This provides a simple way of going either to tape or to disk.

```
9000>CLS :LET od=4:PRINT AT 15,9;"SAVE ROUTINE"
"" Press ""D"" key to save to disk,""" or
""T"" key to save to tape":PAUSE 0
9010 IF INKEY$="t" OR INKEY$="T" THEN LET od=2
```

```
9020 PRINT #od: SAVE "sample.B1" LINE 100
9030 PRINT #od: SAVE "Move .C1"CODE 32000,1000
```

Many of my disks hold Spectrum programs, and have an AUTOSTART menu which has been saved in the Spectrum mode. Naturally, this AUTOSTART menu will not load when one is in the 2068 mode. I append the following lines to the AUTOSTART program. The program is then saved by GOTO 9000. Now, same loading procedure works with both the Spectrum and TS2068 disks.

```
9000>CLEAR 27000:RESTORE 9050
9020 FOR n=23300 TO 23309
9030 READ a: POKE n,a
9040 NEXT n
9050 DATA 205,102,0,62,3,211,244,201,0,0,0
9060 RANDOMIZE USR 23300
9070 RUN
```

Having several drives and a RAMdisk on my system has meant organising my programs somewhat, so as to move back and forth between drives as appropriate with a minimum of effort. I have found it very useful to have the following lines, with appropriate variations, in many of my programs. If you have more than one drive on your system you will find this subroutine very useful.

The "PRINT #4: GOTO 4" points my system to the RAMdisk, where all my program menus are stored. You could modify the routine to point to the appropriate drive. The PRINT #4: NEW causes an AUTOSTART on that drive.

```
720>PRINT AT VAL "15",VAL "6";"Disk copy completed""Press P key for RAMdisk menu, or another key to copy next disk"
```

```
730 PAUSE NOT PI
```

```
740 IF INKEY$<>"P" AND INKEY$<>"p" THEN RUN
```

```
790 PRINT #4: GO TO 4: PRINT #4: NEW
```

SCRABBLE and the LARKEN System by George Chambers

SCRABBLE is a computer game that is patterned very closely after the familiar board game. Among the player options are the facility to SAVE and LOAD a game at any stage of it's progress. Of course the SAVE/LOAD option is to tape. What we want to do here is change the option to save to disk.

The program has a short BASIC component which is used for the LOAD/SAVE procedure. It is this Basic section that we shall modify.

Load the program. Whether you are loading the game from disk or from tape, you should break into it after loading by pressing the Larken NMI button, followed by the A key. The program will not crash. Instead, you will be able to list it.

Modify the listing to correspond to the listing below. Take note of the following points:

1. The N\$ has been DIMmed to n\$(10) in the program. Change this by entering as a direct command... DIM n\$(9). (don't use the commands CLEAR or RUN)

2. I had to instal line 5 because I found the program would not operate on my choice of instruction, "LET N\$ = N\$ + ".Cs". That is why the LOAD and SAVE commands use the term n\$+d\$ for the program name. I can't explain it.

3. Do not elaborate on the BASIC program. The game code is very low in memory and there is a danger you may overwrite it.

Re-enter the game by GOTO 10, then immediately SAVE the modified program to disk by using the NMI function. Rename the program with a RENAME facility.

```
4 INPUT "FILENAME : "; LINE n$: IF LEN n$>6
THEN LET n$=n$( TO 6)
5 LET d$=".Cs": RETURN
10 GO TO USR VAL "46474"
50 BORDER VAL "5": CLS : PRINT AT VAL "6",PI*
PI;"Loading": GO SUB PI: PRINT USR VAL "100": L
OAD n$+d$CODE : GO TO USR VAL "46477"
90 GO TO USR VAL "46474"
100 BORDER VAL "5": CLS : PRINT AT VAL "6",PI*
PI;"Saving game": GO SUB PI: PRINT USR VAL "100"
: SAVE n$+d$CODE VAL "60368",VAL "5167": GO TO
USR VAL "46477"
```

PROJECTS FOR THE TS COMPUTERS

Beginning with this issue, we would like to present a column for those of you who are interested in building or modifying hardware for your TS1000 or TS2068. Most of the projects lined up will be directed at the TS1000 but instructions or modifications will be included for the TS2068. Some of the ideas for presentation are old, some are borrowed and we will do our best to give credit where it is due. Don't be surprised if you see IBM or APPLE pop up once in a while.

Projects currently being considered include:

- | | |
|-------------------------|---------------|
| 1. Sound Generator | TS1000 |
| 2. Eprom Burner | TS1000/TS2068 |
| 3. Parallel Interface | TS1000/TS2068 |
| 4. MIDI Interface | TS1000/TS2068 |
| 5. 64k SRAM Board | TS1000 |
| 6. Keyboard Interface | TS1000/TS2068 |
| 7. Buffered Motherboard | TS1000/TS2068 |

In addition we hope to be able to present VERO/Proto Board layouts for projects that have appeared in this newsletter as well as in others as an aid to those people who have been hesitant in constructing hardware for their computers. To assist us in this endeavour, we would like to correspond with anyone who has put a project together and would like to share the fruits of their labour.

We would like to invite your questions and suggestions for material to be incorporated into this column.

PROJECT No. 1 "Mini Mother Board"

There are two accepted ways of attaching peripheral devices to the TS computers. Anyone who owns a Gladstone or Memotech RAM Pack and a thermal printer is familiar with the Feed-through Connector that these devices have. While they are inexpensive they do have the disadvantage of flexibility (where none is wanted). The second type of connector that is used is a "mother board/daughter board" arrangement that is used in larger computer systems. The previously mentioned IBM and APPLE systems use this method.

Maplin from England and John Oliger in the U.S.A. offer motherboards for sale. The base price for a bare printed circuit is about \$10 to \$15. If your familiar with the surplus market you can often find something suitable at very reasonable prices.

A one slot motherboard can be constructed from an APPLE or IBM Extender Board. Prices range from \$5 to \$10 depending on the computer store where you buy your parts. The former is suitable for the TS1000 (50 conductors) and the latter is almost perfect for the TS2068 (62 conductors).

PARTS: Extender Board
2 edge connector (wire wrap)
0.100 inch pitch; 50 (or 60) pins

CONSTRUCTION:

1. Trim the edge connectors down in length to fit the buss on the back of your computer.
2. Bend and trim the leads on one of the connectors to about 1/4 inches. the leads have to fit in the double row of plated holes on the extender board.
3. Bend the leads of the other connector to fit neatly over the end of the extender board.
4. Cut the extender board shorter if required and solder the two edge connectors onto the board ensuring that they are square. Solder two leads first, make your adjustments, then solder leads on alternate sides. You can make adjustments by reheating the solder joints.
5. Clean the flux off the board and check for solder bridges.
6. Test the board by inserting it between your computer and a memory pack or printer. If the cursor does not appear or if nothing prints then recheck your work.

by Rene Bruneau

With an increasing number of TS2068 owners getting disk drives it is natural enough to consider the creation of a Larken disk library. I would like to ask for comments on the subject from other owners.

Personally, I have mixed feelings on the matter. Although I am in sympathy with the idea I am also aware that it would devolve on me (probably) to maintain and administer it!! This has a rather inhibiting effect, and possibly this affects my thoughts on the matter.

Let me say that I am basically enthusiastic about the idea. Our club has been in the forefront of the Larken Disk system movement and it seems only right and proper that we should start a Larken disk library. However lets consider the ramifications of it. Lets ask some questions.

1. What programs would be on the disks. Would they be simply a copy of the existing tape library. Or should they be confined to programs which are particularly appropriate to the Larken system.
2. Should it support more than one disk format or be confined to one agreed-on format.
3. If a single format is determined to be appropriate, which should it be.
4. What about the Larkenized Oliger and Aerco systems. How would they be handled.

Enough questions. Let me give you some of my own thoughts:

Firstly, it should be simple to administer. This probably means that it should not duplicate the programs presently in our tape library. Tapes probably will continue to be our common medium of exchange, and I think the disks should contain those programs which are unique to the disk system.

The documentation for the programs should be on the disks containing the program. These could be in the form of Tasword files which could be viewed on screen or printed out on a large printer, using the new version 3 Larken Eprom DOS.

Ideally, a disk should contain only one program or a suite of related programs, not an accumulation of unrelated materials, which may already exist on club library tapes. It goes without saying that the disks would have only material released to the public domain.

Probably the Master disks should be in one format, namely DSDD. Members who could not handle that format could request a copy in the desired format, i.e. Single-sided, or Quad density.

It appears most appropriate to simply mail out disks, without return. This because the mailing costs currently seem to equate to the cost of the disk itself. There would have to be a nominal reimbursement to cover disk purchases and postage costs.

In my experience in writing Larken utilities, it is a constant task, upgrading and adding new features to them. This inclines me to feel that a disk should be identified to the member author so that he could answer questions about it, and incorporate any improvements which may be suggested or become evident. This, rather than expecting the disk librarian to do it, or worse, improvements not being done at all.

Larken owners, let's hear from you. Your comments/suggestions are welcomed.

The following is a listing that Larry Kenny sent me, to be used with the Ed Grey RS232C Interface (Z8SI/O).

This interface consists of two boards and is designed to work with the TS2068. It is now available only as bare boards, for US\$24.50 , plus \$2.50 S&H.

For more information on this, and many other TS2068 accessories, send a stamped self-addressed envelope, plus \$1 to: Ed Grey Enterprises, P.O. Box 2186, Inglewood, CA 90305.

I suggest that Canadian T/S users send a Postal Money Order in US Funds in lieu of stamps or cheques.

G. Chambers

```
10 REM RS232C DRIVER FOR LKDOS                      Versio
n 3 EPROM
15 RANDOMIZE USR 100: OPEN #4,"dd"
20 LET ST=191: REM 8251 Status
22 LET DT=159: REM 8251 Data
25 PRINT "Select Parameters"
30 PRINT "1 8/1/N at 300 baud" "2 7/1/E
at 300 Baud" "3 8/1/N at 1200 Baud" "4 7/1/
E at 1200 Baud"
40 INPUT "Select 1 - 4 ";a
50 IF a=1 THEN LET baud=111
60 IF a=2 THEN LET baud=123
70 IF a=3 THEN LET baud=110
80 IF a=4 THEN LET baud=122
90 IF a>4 THEN GO TO 40
100 REM Initialize RS232C Port
110 OUT ST,0: OUT ST,0: OUT ST,0: OUT ST,64: OU
T ST,baud: OUT ST,183
120 REM Install Driver in LKDOS Cartridge using
PRINT #4,POKE
130 FOR a=16100 TO 16109
140 READ v: PRINT #4: POKE a,v
150 NEXT a
160 PRINT #4: POKE 8216,16100
170 PRINT #4: POKE 16096,4
180 PRINT #4: OPEN #3,"lp"
200 DATA 219,ST,203,87,40,250,241,211,DT,201
8999 STOP
9000 CLEAR : PRINT USR 100: SAVE "driver.B1"
```

GOT ANY QUESTIONS?

GOT ANY ANSWERS?

SEND THEM TO

SINK-LINK AND

WE'LL PUBLISH

THEM FOR ALL

MEMBERS TO USE

AND COMMENT ON.

REVIEW

FIRST CLASS FONTS (BYTE POWER)

The past couple of newsletter covers, as well as this one, have examples of some of the many fonts available to the user in Byte Power's "First Class Fonts".

This package contains 17 font styles (see fig. 1) with the option to print normal or double height characters (fig. 2).

In order to use a font in your own program, you must first save the font code of your choice (as per instructions, noting start address and length) then incorporate the the call out routine (again, as per instructions) into your Basic listing.

The instructions are laid out simply enough that with a little experimenting you'll soon get the results you want. The listing at right is what I created to produce the cover blurb in the futuristic font with the 2040 printer. It's really quite simple (can anyone but there make it more elegant? please? -ed)

This package also contains 3 more programs. The next one produces an old English character set (fig. 3). The third program is a 64 column print utility and the last program is a character set definer. Not a bad deal in one package. For more information and current prices contact: Byte Power, 1748 Meadowview Avenue, Pickering, Ontario, L1V 3G8

Jeff Taylor

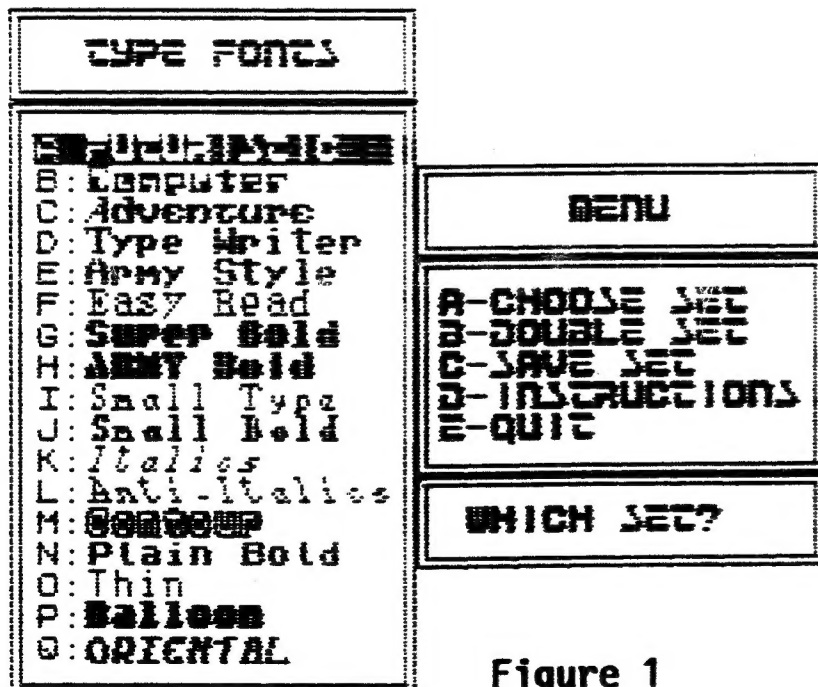


Figure 1

SINC-LINK IS A PUBLICATION OF

SINC-LINK IS A PUBLICATION OF

Figure 2

1 REM Newsletter Blurb
by Jeff Taylor
Toronto Timex-Sinclair
Users Club

```

2 REM
5 FOR N=0 TO 20 STEP 2
10 READ A$
15 IF A$="Z" THEN GO TO 90
16 IF A$="" THEN COPY
20 LET A=43460-256
30 POKE 23606,A-256*INT (A/256)

40 POKE 23607,INT (A/256)
41 LET X=N
42 LET Y=0
50 PRINT AT X,Y:A$
60 LET A=A+768
70 POKE 23606,A-256*INT (A/256)

80 POKE 23607,INT (A/256)
85 PRINT AT X+1,Y:A$
86 NEXT N
87 COPY
88 CLS : IF X=20 THEN GO TO 5
89 GO TO 10
90 POKE 23606,0
100 POKE 23607,60
210 DATA "SINC-LINK IS A PUBLICATION OF"
211 DATA "THE TORONTO TIMEX-SINCLAIR USERS"
212 REM DATA "CLUB AND IS ISSUED 6 TIMES A"
213 DATA "YEAR. COPIES OF THE NEWSLETTER"
214 DATA "ARE $1.50 EACH FOR NON-MEMBERS."
215 DATA "CLUB MEMBERS RECEIVE A FREE COPY"
216 DATA "AS PART OF THE $20.00 ANNUAL"
217 DATA "MEMBERSHIP FEE."
218 DATA " "
219 DATA " "
220 DATA "NEWSLETTERS ARE EXCHANGED, FREE"
221 DATA "OF CHARGE, WITH OTHER TIMEX-"
222 DATA "SINCLAIR USER GROUPS."
223 DATA " "
224 DATA " "
225 DATA "ALL MATERIAL IS PUBLIC DOMAIN"
226 DATA "AND CAN BE REPRINTED. PLEASE"
227 DATA "CREDIT THIS PUBLICATION AND THE"
228 DATA "AUTHOR IF YOU COPY MATERIAL."
229 DATA " "
230 DATA " "
231 DATA "SEND CORRESPONDANCE TO:"
232 DATA " "
233 DATA "Attention: SINC-LINK EDITORS,"
234 DATA "TORONTO TIMEX-SINCLAIR USERS"
235 DATA "CLUB, 14 RICHMOND COURT,"
236 DATA "SCARBOROUGH, ONTARIO, CANADA,"
237 DATA "M1K 2Y1"
238 DATA " "
239 DATA "Z"

```

Figure 3

OLD ENGLISH

A lot of my Timex club work involves writing letters in duplicate. That is to say, I make a carbon copy. As a consequence, most of the time I use single sheets of paper, rather than formfeed. With a lengthy letter this gets quite involved. What I required was a modification to Tasword to enable it to PAUSE after printing out a page, so that I could insert the next page in the printer.

When Larry K's new version 3 EPROM DOS came out with it's improved printer driver routine it seemed an appropriate time to rework the Tasword print routine.

The results of this exercise are shown in the following listing.

Several things should be noted in this listing:

1. There are a number of variables in the listing which are undeclared. They are variables with the names oo, oa, ob, oc, od, etc. The idea is to create a set of easily identifiable variables names.

Bill Jones, editor of the Timex periodical, TS UPDATE, must be given credit for this idea. Bill's standard is to make oo=0, oa=1, ob=2, oc=3, od=4, etc., continuing up to ot=20. Of course, for a particular program you may not need every one of the. Use only those that you need. But stay with the philosophy, and at any time it is easy to determine what a variable represents, by counting it out alphabetically.

I have used this system of variables throughout Tasword to save space, so I declared the variables very early in the program. You may do likewise, or simply declare them in a new line 202.

2. This listing gives a number of options, such as Drive selection, page pause, lines per page, starting line #, line spacing, etc. Default values are given, so that if they are acceptable you may simply press the ENTER key. I entered "buried" colour codes for each of the INPUT queries, to make them stand out. These do not appear in the listing, but you might consider their use.

3. Line 203 sets up the Larken Printer driver. Poke 16090 establishes page length (64 characters, Poke 16092 establishes whether there is to be a line feed along with the carriage return (no, in the case of my printer, yours might be different)

You may note that the LINE 203 that invokes the Larken printer driver has been treated as a REM line. This is because I later realized that to use the Tasword "graphics" printer codes I would have to stay with the Tasword printer driver code!!

The Page Pause routine operates on this basis. When the feature is invoked at line 205, lines 230 and 265 mark out the data that is to be printed on the initial sheet. Printing is then carried out by lines 275, 280, and 285. The program then jumps to line 287 to print a "next page" message, and to introduce a PAUSE until a key is pressed.

When a key is pressed line 287 continues, through to line 289 to mark the next block of data that is to be printed; jumps to line 275 for a repeat of the printing routine; and so on until the last of the text that is to be printed is reached. Lines 266 & 288 check for the end of the file and make any necessary correction so that the printer stops promptly after the last line of text has been printed. That is to say, so that it does not sweep over empty lines to the end of the marked out page.

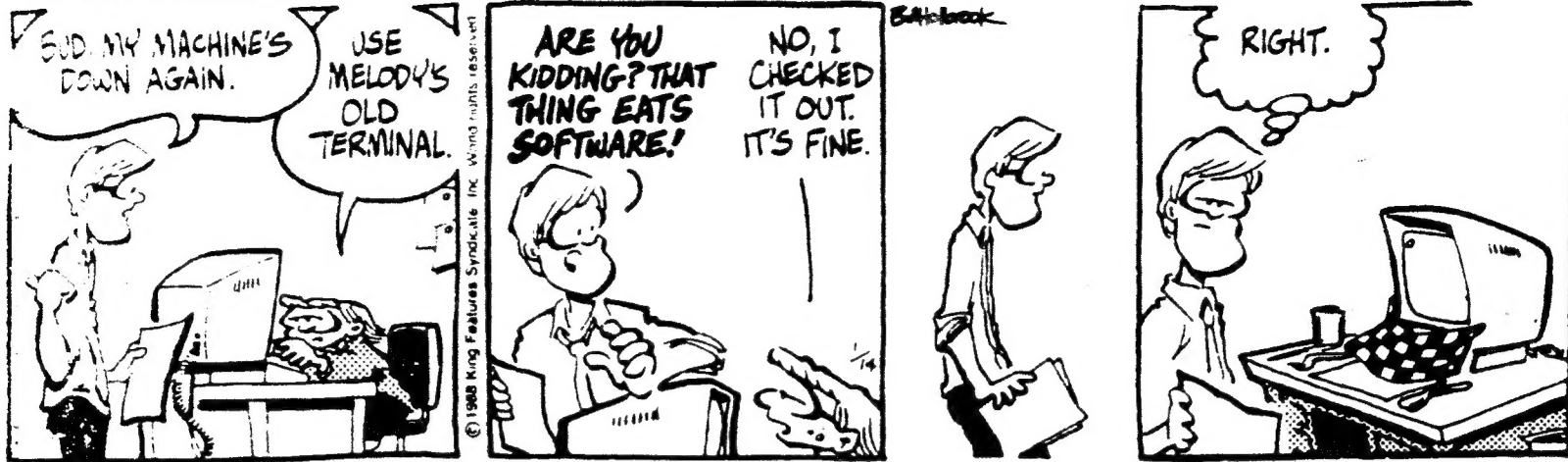
Modified TASWORD (Partial listing)

```

200 CLS : PRINT AT VAL "14",VAL "10";"PRINT OP
TIONS"; PRINT " just press ENTER for default
values given in brackets"
201 DIM d$(32)
203 REM For Larken Printer Driver: REM PRINT
#od: OPEN #oc,"lp": PRINT #od: POKE VAL "16090"
,VAL "65": PRINT #od: POKE VAL "16092",oo
205 LET h=oo: LET p=oa: POKE VAL "23658",oo: I
NPUT " Page Pause? N/(Y)";a$: IF a$="n
" THEN LET p=oo: LET h=a: GO TO VAL "220"
212 INPUT " Lines per page?(55)";b$: IF b$=
"" THEN LET b$="55": LET g=VAL "3520": GO TO V
AL "220"
215 LET g=VAL b$*VAL "64"
220 POKE VAL "60927",oa: REM left margin
222 INPUT " Line spacing? (1)";a$: IF a$=""
THEN LET a$="1"
223 POKE VAL "62235",VAL a$
225 INPUT " Start at line? (1)";a$: IF a$=""
THEN LET a$="1"
230 LET c=VAL "64"*(INT VAL a$-VAL "1"); LET b
=c+FN p(VAL "62216"); LET x=VAL "60045": GO SUB
VAL "950"
240 INPUT " Finish at line? (last) ";a$: IF
a$="" THEN LET b=a-c: GO TO VAL "250"
245 LET b=VAL "64"*INT VAL a$-c
250 RANDOMIZE USR VAL "59806": RANDOMIZE USR (
FN p(VAL "62472"))
260 CLS : PRINT AT VAL "20",oo;"Press the q ke
y to quit printing"
265 IF p THEN LET b=VAL b$*VAL "64": LET g=b:
LET h=b
266 IF g>a THEN LET b=a: LET g=b: LET h=b
270 LET x=VAL "60049": GO SUB VAL "950"
275 LET c=PEEK VAL "62470": IF c<>oo THEN LPR
INT CHR$ c
280 RANDOMIZE USR VAL "60038"
282 IF h>a THEN GO TO VAL "290"
285 LET c=PEEK VAL "62471": IF c<>oo THEN LPR
INT CHR$ c
286 IF PEEK VAL "23560"=VAL "113" THEN GO TO
290
287 IF p AND h<a THEN PRINT AT VAL "15",oa;"P
ress a key for next page"" Lines left "; PA
PER of;(a-h)/VAL "64"; PAPER og: PAUSE oo: LET
b=FN p(VAL "60045"); LET b=b+VAL b$*VAL "64": L
ET x=VAL "60045": GO SUB VAL "950": LET h=h+g
288 IF h>a THEN LET b=a-(h-g): LET x=VAL "60
049": GO SUB VAL "950"
289 PRINT AT VAL "15",oa;d$d$: GO TO VAL "275
"
290 RANDOMIZE USR VAL "59806": GO TO VAL "10"

```

On The Fastrack



Postmaster, If Undelivered Return To:

Toronto Timex—Sinclair Users Club
14 Richome Court, Scarborough,
Ontario, M1K 2Y1, CANADA

